In the Claims:

Claims 1, 4, 5 and 33 are amended.

Claims 1, 4-17, 20, 21, 23-27, 29, and 32-37 remain pending and are listed below:

1.   (Currently Amended) A command line utility embodied in one or more computer-readable storage media, the command line utility comprising:

a command schema including one or more commands for engaging, from a management station on a network, an operating system on at least one selected target station accessible over the network enabling at least one of retrieval of management information from and initiation of a management service available through an object model target schema recognized by the at least one selected target station accessible over the network, the command schema comprising:

an alias class which defines a command template, each alias of the alias class representing a single command;

a verb class ~~as a subclass to the alias class~~, each instance of the verb class representing a behavior available through an alias;

a format class ~~as a subclass to the alias class~~, each instance of the format class representing a list of properties that will be returned through processing an alias; and

a connection class ~~as a subclass to the alias class~~, each instance of the connection class providing parameters used through an alias when establishing a connection to a target namespace within the target schema;

an interactive user interface configured to receive the one or more commands in the command schema from a user of the management station,

allowing the user to formulate via the command line utility an instant request for at least one of management information and management services and communicate a response of the at least one selected target station to the user; and

an object model command schema to define a mapping between the one or more received commands in the command schema and the object model target schema and interpret the one or more commands from the command schema into a management command operable to cause one of the retrieval of the management information from and the initiation of the management service on the at least one selected target station.

2.   (Canceled).

3.   (Canceled).

4.   (Currently Amended) A command line utility as recited in claim 1, the command schema further comprising:

a parameter class ~~as a subclass to the verb class~~, each instance of the parameter class representing parameters accepted by a verb from the verb class.

5.   (Currently Amended) A command line utility as recited in claim 1, the command schema further comprising:

a property class ~~as a subclass to the format class~~, each instance of the property class representing property values for properties within a list of properties.

6.   (Previously Presented) A command line utility as recited in claim 1, the command schema further comprising:

a localized string class, each instance of the localized string class providing language-specific text relevant to an alias.

7.   (Previously Presented) A command line utility as recited in claim 1, the command schema further comprising:

a qualifiers class, each instance of the qualifiers class representing qualifiers to be applied to elements of an alias.

8.   (Previously Presented) A command line utility as recited in claim 1, the command schema further comprising:

a see-also association, each instance of the see-also association providing associations to aliases which are related to an alias.

9.   (Original) A command line utility as recited in claim 1, wherein the one or more commands are organizable into role-oriented namespace, such that commands related to particular administrative tasks are found together.

10.   (Original) A command line utility as recited in claim 1, wherein the command schema is extensible to permit the generation of additional commands to be added to the one or more commands.

11.   (Original) A command line utility as recited in claim 1, wherein the command schema is configurable to permit reconfiguration of the one or more commands.

12.   (Previously Presented) A command line utility as recited in claim 1, the command schema further comprising:

a parameter class to define parameters accepted by a verb instance;

a property class to define property instances, each property instance representing a property value from a property list;

a qualifier class to define qualifier instances, each qualifier instance representing constraints on elements of an alias instance;

a localized string class to define localized string instances, each localized string instance representing a text localization for translating text into a localized language; and

a see-also association to associate an alias instance with other related alias instances.

13.   (Original) A command line utility as recited in claim 1, wherein the target schema comprises a WMI object model.

14.   (Original) A command line utility as recited in claim 1, wherein the command schema is a plurality of command schemas, each command schema representing an instant set of one or more commands.

15.   (Original) A command line utility as recited in claim 1, wherein the command schema resides on a local machine and the target schema resides on one or more remote machines.

16.   (Previously Presented) A management command system operable to retrieve management information and initiate a management service available through an object model target schema recognized by at least one selected target station accessible over a network, the management command system comprising:

an interactive user interface configured to receive one or more commands in the command schema formulated as an instant request by a user at a management station for engaging an operating system on at least one selected target station and communicate a response of the at least one selected target station to the user; and

a control schema including the one or more commands to be received by the interactive user interface, including:

an alias class to define alias instances, each alias instance representing the at least one command;

a verb class to define verb instances, each verb instance representing behavior available through an alias instance;

a parameter class to define parameters accepted by a verb instance;

a format class to define format instances, each format instance having a list of properties to be displayed through an alias instance;

a property class to define property instances, each property instance representing a property value from a property list;

a connection class to define connection instances, each connection instance representing connection parameters used by an alias instance to establish a connection to the target schema;

a qualifier class to define qualifier instances, each qualifier instance representing constraints on elements of an alias instance;

a localized string class to define localized string instances, each localized string instance representing a text localization for translating text into a localized language; and

a see-also association to associate an alias instance with other related alias instances.

17. (Previously Presented) A management application embodied in one or more computer-readable storage media, the management application comprising:

a set of commands generated by an object model command schema to operate against an object model target schema of at least one selected target, the command schema defining a mapping between the set of commands and the target schema, and wherein the command schema is configurable to permit reconfiguration of individual commands in the set of commands; and

an interactive interface utility configured to allow a user to formulate an instant request and receive the user command to facilitate implementation of individual commands within the set of commands, wherein the interactive interface utility includes at least one of:

a command line interface utility configured to receive textual entry of the user command; and

a graphical user interface utility.

18.   (Canceled).

19.   (Canceled).

20.   (Original) A management application as recited in claim 17, wherein the set of commands is organizable into role-oriented namespace, such that individual commands related to particular administrative tasks are found together.

21.   (Original) A management application as recited in claim 17, wherein the command schema is extensible to permit the generation of additional commands to be added to the set of commands.

22.   (Canceled).

23.   (Original) A management application as recited in claim 17, wherein the command schema comprises:

an alias class to define alias instances, each alias instance representing a command within the set of commands;

a verb class to define verb instances, each verb instance representing behavior available through an alias instance;

a parameter class to define parameters accepted by a verb instance;

a format class to define format instances, each format instance having a list of properties to be displayed through an alias instance;

a property class to define property instances, each property instance representing a property from a property list;

a connection class to define connection instances, each connection instance representing connection parameters used by an alias instance to establish a connection to the target schema object model;

a qualifier class to define qualifier instances, each qualifier instance representing constraints on elements of an alias instance;

a localized string class to define localized string instances, each localized string instance representing a text localization for translating text into a localized language; and

a see-also association to associate an alias instance with other related alias instances.

24.   (Previously Presented) A management application embodied in one or more computer-readable storage media, the management application comprising:

a first object model to control configuration and behavior of the management application in operating against and managing an operating system of a second object model operating on at least one selected target system accessible over a network, the first object model including:

a set of commands for directing the configuration and the behavior of the management application; and

a command schema defining a mapping between the set of commands and instructions understandable by the second object model; and

an interactive interface configured to receive the set of commands from a user at a management station, wherein the interactive interface includes at least one of:

a command line interface utility configured to allow the user to formulate an instant command and receive textual entry of the user command; and

a graphical user interface utility.

25.   (Original) A management application as recited in claim 24, wherein the first object model comprises:

an alias class to define alias instances, each alias instance representing a command used by the management application;

a verb class to define verb instances, each verb instance representing behavior available through an alias instance;

a parameter class to define parameters accepted by a verb instance;

a format class to define format instances, each format instance having a list of properties to be displayed through an alias instance;

a property class to define property instances, each property instance representing a property from a property list;

a connection class to define connection instances, each connection instance representing connection parameters used by an alias instance to establish a connection to the second object model;

a qualifier class to define qualifier instances, each qualifier instance representing constraints on elements of an alias instance;

a localized string class to define localized string instances, each localized string instance representing a text localization for translating text into a localized language; and

a see-also association to associate an alias instance with other related alias instances.

26.   (Original) A management application as recited in claim 24, wherein the second object model comprises a WMI object model.

27.   (Previously   Presented)   A   computer-implemented   method comprising:

receiving a command from a user that was formulated as an instant request by the user at a management station through an interactive command line interface;

fetching an alias for the command, wherein the alias is an instance of an alias class and is the primary class in a command schema, and wherein the command schema additionally includes:

a verb class to define verb instances, each verb instance representing behavior available through an alias;

a parameter class to define parameters accepted by a verb instance;

a format class to define format instances, each format instance having a list of properties to be displayed through an alias;

a property class to define property instances, each property instance representing a property from a property list;

a connection class to define connection instances, each connection instance representing connection parameters used by an alias to establish a connection to the target namespace;

a qualifier class to define qualifier instances, each qualifier instance representing constraints on elements of an alias;

a localized string class to define localized string instances, each localized string instance representing a text localization for translating text into a localized language; and

a see-also association to associate an alias with other related aliases;

interpreting the command based on the alias and the current operating environment of the command line interface;

executing the command as one or more WMI API calls against a targeted namespace representing at least one selected target system that is accessible via a network;

receiving WMI data in XML form;

applying an XSL style sheet format the WMI data; and

presenting the WMI data through the command line interface.

28. (Canceled).

29. (Previously Presented) A method as recited in claim [[28]] 27, wherein the alias and the current operating environment are determined by command entries to the command line interface.

30. (Canceled).

31. (Canceled).

32. (Original) One or more computer-readable media storing computer-executable instructions that, when executed on one or more processors, perform the method of claim 27.

33. (Currently Amended) A computer-implemented method of managing objects in a target schema comprising:

providing an interactive user interface configured to allow a user to formulate a set including one or more instant commands operable to receive management information and initiate a management service and receive the commands from the user;

defining a command structure through an object-oriented command schema, the command schema comprising:

an alias class which defines a command template, each alias of the alias class representing a single command;

a verb class ~~as a subclass to the alias class~~, each instance of the verb class representing a behavior available through an alias;

a format class ~~as a subclass to the alias class~~, each instance of the format class representing a list of properties that will be returned through processing an alias; and

a connection class ~~as a subclass to the alias class~~, each instance of the connection class providing parameters used through an alias when establishing a connection to a target namespace within the target schema;

instantiating an object of the alias class as an alias by receiving parameters of the alias class via the commands received from the user through the user interface, the alias representing a command which maps to an object in the target schema maintained on at least one selected target system accessible via a network configured to at least one of retrieve management information and initiate a management service; and

executing the command against the object in the target schema of the at least one selected target system.

34. (Previously Presented) A computer-implemented method as recited in claim 33, further comprising:

receiving data related to the object in the target schema in XML form;

retrieving an XSL file from a location defined through the alias;

formatting the data for display by applying an XSL style sheet from the XSL file; and

displaying the data through the user interface.

35. (Previously Presented) A computer-implemented method as recited in claim 33, wherein the user interface is a command line user interface.

36. (Previously Presented) A computer-implemented method as recited in claim 33, wherein the user interface is a graphical user interface.

37.   (Previously Presented) One or more computer-readable media storing computer-executable instructions that, when executed on one or more processors, perform the computer-implemented method of claim 33.